

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (currently amended) A method for managing packets in a shared memory buffer that serves a plurality of output links comprising:
  - writing packets into said shared memory buffer;
  - linking said packets into a linked list in the order that said packets are written into said shared memory buffer;
  - using said linked list to identify the oldest packet that is intended for an output link that is available for packet transmission; ~~and~~
  - dispatching, to the respective available output link, said identified oldest packet that is intended for an available output link even if said identified oldest packet is preceded by an older packet that is intended for an unavailable output link; and
  - establishing an output link identifier (ID) for each packet that identifies the intended output link of the respective packet.
2. (original) The method of claim 1 wherein linking said packets into a linked list includes establishing a NEXT pointer for each packet that identifies the next packet on the linked list.
3. (canceled)
4. (currently amended) The method of claim ~~13~~ further including using said output link ID of a packet to determine whether the respective packet is intended for an available output link.
5. (original) The method of claim 1 further including establishing a linked list HEAD pointer that identifies the oldest packet that is written into said shared memory buffer.

6. (original) The method of claim 5 wherein identifying the oldest packet that is intended for an available output link is repeated in dispatch cycles, wherein each dispatch cycle begins at the packet that is identified by the linked list HEAD pointer.
7. (original) The method of claim 5 wherein dispatching said identified oldest packet includes:
  - if said dispatched packet is at the head of said linked list, then setting said linked list HEAD pointer to identify the next packet on said linked list; and
  - if said dispatched packet is not at the head of said linked list, then deleting said dispatched packet from said linked list by setting a NEXT pointer for the previous packet on said linked list to the same value as the dispatched packet's NEXT pointer.
8. (original) The method of claim 1 further including identifying the availability of each of said output links.
9. (original) The method of claim 8 wherein packets are dispatched in dispatch cycles and further including assessing the availability of each of said output links at the beginning of each dispatch cycle.
10. (original) The method of claim 9 further including dispatching multiple packets in the same dispatch cycle if there are multiple packets that are intended for available output links.
11. (original) The method of claim 1 wherein said packets are variable-length packets.

12. (currently amended) A system for managing packets in a shared memory buffer that serves a plurality of output links comprising:

a shared memory buffer, that serves a plurality of output links, for storing packets that are intended for said output links;

a buffer controller connected to manage packets stored within said shared memory buffer, said buffer controller including means for:

linking said packets into a linked list in the order that said packets are written into said shared memory buffer;

using said linked list to identify the oldest packet that is intended for an output link that is available for packet transmission; and

dispatching, to the respective available output link, said identified oldest packet that is intended for an available output link even if said identified oldest packet is preceded by an older packet that is intended for an unavailable output link;

wherein said buffer controller further includes means for establishing an output link identifier (ID) for each stored packet that identifies the intended output link of the respective packet.

13. (original) The system of claim 12 wherein linking said packets into a linked list includes establishing a NEXT pointer for each stored packet that identifies the next packet on the linked list.

14. (canceled)

15. (currently amended) The system of claim ~~12~~14 wherein said buffer controller further includes means for using said output link ID of a packet to determine whether the respective packet is intended for an available output link.

16. (original) The system of claim 12 wherein said buffer controller further includes a linked list HEAD pointer that identifies the oldest packet that is written into said shared memory buffer.

17. (original) The system of claim 16 wherein identifying the oldest packet that is intended for an available output link is repeated in cycles, wherein each cycle begins at the packet that is identified by the linked list HEAD pointer.

18. (original) The system of claim 16 wherein said buffer controller includes means for:

if a dispatched packet is at the head of said linked list, then setting said linked list HEAD pointer to identify the next packet on said linked list; and

if a dispatched packet is not at the head of said linked list, then deleting said dispatched packet from said linked list by setting a NEXT pointer for the previous packet on the list to the same value as the dispatched packet's NEXT pointer.

19. (original) The system of claim 12 further including connections between said buffer controller and said output links for identifying the availability of each of said output links.

20. (original) The system of claim 19 wherein packets are dispatched in dispatch cycles and wherein the availability of each of said output links is assessed in each dispatch cycle.

21. (original) The system of claim 20 wherein said buffer controller includes means for dispatching multiple packets in the same dispatch cycle if there are multiple packets that are intended for available output links.

22. (original) The system of claim 12 wherein said packets are variable-length packets.

23. (currently amended) A method for managing packets in a shared memory buffer that serves a plurality of output links comprising:

writing packets into said shared memory buffer;

linking said packets into a linked list in the order that said packets are written into said shared memory buffer;

identifying the oldest packet that is written into said shared memory buffer;

identifying the availability of each of said plurality of output links;

starting at the identified oldest packet that is written into said shared memory buffer, examining said packets in said output buffer in the order that is dictated by said linked list to identify the oldest packet that is intended for an available output link; ~~and~~

dispatching, to the respective available output link, the packet that is identified as the oldest packet that is intended for an available output link; and

setting an output link identifier (ID) for each packet that identifies the intended output link of the respective packet.

24. (original) The method of claim 23 wherein said oldest packet that is intended for an available output link is dispatched even if said dispatched oldest packet is not identified as the oldest packet on said linked list.

25. (original) The method of claim 23 wherein linking said packets includes setting a NEXT pointer for each packet that identifies the next packet on said linked list.

26. (original) The method of claim 25 further including establishing a linked list HEAD pointer that identifies the oldest packet that is written into said shared memory buffer.

27. (original) The method of claim 26 wherein dispatching said oldest packet includes:

if said dispatched oldest packet is at the head of said linked list, then setting said linked list HEAD pointer to identify the next packet on said linked list; and

if said dispatched oldest packet is not at the head of said linked list, then deleting said oldest packet from said linked list by setting the NEXT pointer for the previous packet on said linked list to the value of the dispatched packet's NEXT pointer.

28. (canceled)

29. (currently amended) The method of claim ~~23~~<sup>28</sup> wherein examining said packets to identify the oldest packet that is intended for an available output link includes accessing said output link ID of a packet to determine the intended output link of the respective packet.

30. (original) The method of claim 23 wherein packets are dispatched in dispatch cycles, said method further including assessing the availability of said output links at the beginning of each dispatch cycle.

31. (original) The method of claim 30 further including dispatching multiple packets in a single dispatch cycle.

32. (currently amended) A system for managing packets in a shared memory buffer that serves a plurality of output links comprising:

a shared memory buffer, that serves a plurality of output links, for storing packets intended for said output links:

a buffer controller connected to manage packets stored within said shared memory buffer, said buffer controller including means for:

linking said packets into a linked list in the order that said packets are written into said shared memory buffer;

identifying the oldest packet that is written into said shared memory buffer;

identifying the availability of each of said plurality of output links;

starting at the identified oldest packet that is written into said shared memory buffer, examining said packets in said output buffer in the order that is dictated by said linked list to identify the oldest packet that is intended for an available output link; and

dispatching, to the respective available output link, the packet that is identified as the oldest packet that is intended for an available output link;

wherein said buffer controller further includes means for setting an output link identifier (ID) for each packet that identifies the intended output link of the respective packet.

33. (original) The system of claim 32 wherein said oldest packet that is intended for an available output link is dispatched even if said dispatched oldest packet is not identified as the oldest packet on said linked list.

34. (original) The system of claim 32 wherein linking said packets includes setting a NEXT pointer for each packet that identifies the next packet on said linked list.

35. (original) The system of claim 34 wherein said buffer controller further includes means for establishing a linked list HEAD pointer that identifies the oldest packet that is written into said shared memory buffer.

36. (original) The system of claim 35 wherein said buffer controller includes means for:

if said dispatched oldest packet is at the head of said linked list, then setting said linked list HEAD pointer to identify the next packet on said linked list; and

if said dispatched oldest packet is not at the head of said linked list, then deleting said oldest packet from said linked list by setting the NEXT pointer for the previous packet on said linked list to the value of the dispatched packet's NEXT pointer.

37. (canceled)

38. (currently amended) The system of claim ~~32~~<sup>37</sup> wherein examining said packets to identify the oldest packet that is intended for an available output link includes accessing said output link ID of a packet to determine the intended output link of the respective packet.

39. (original) The system of claim 32 further including connections between said buffer controller and said output links for identifying the availability of each of said output links.

40. (original) The system of claim 32 wherein said buffer controller includes means for dispatching packets in cycles and for assessing the availability of said output links at the beginning of each dispatch cycle.

41. (original) The system of claim 40 wherein said buffer controller further includes dispatching multiple packets in a single dispatch cycle.

42. (new) A method for managing packets in a shared memory buffer that serves a plurality of output links comprising:

writing packets into said shared memory buffer;

linking said packets into a linked list in the order that said packets are written into said shared memory buffer;

using said linked list to identify the oldest packet that is intended for an output link that is available for packet transmission; and

dispatching, to the respective available output link, said identified oldest packet that is intended for an available output link even if said identified oldest packet is preceded by an older packet that is intended for an unavailable output link;

wherein linking said packets into a linked list includes establishing a NEXT pointer for each packet that identifies the next packet on the linked list.

43. (new) A system for managing packets in a shared memory buffer that serves a plurality of output links comprising:

a shared memory buffer, that serves a plurality of output links, for storing packets that are intended for said output links;

a buffer controller connected to manage packets stored within said shared memory buffer, said buffer controller including means for:

linking said packets into a linked list in the order that said packets are written into said shared memory buffer;

using said linked list to identify the oldest packet that is intended for an output link that is available for packet transmission; and

dispatching, to the respective available output link, said identified oldest packet that is intended for an available output link even if said identified oldest packet is preceded by an older packet that is intended for an unavailable output link;

wherein linking said packets into a linked list includes establishing a NEXT pointer for each stored packet that identifies the next packet on the linked list.

44. (new) A method for managing packets in a shared memory buffer that serves a plurality of output links comprising:

writing packets into said shared memory buffer;

linking said packets into a linked list in the order that said packets are written into said shared memory buffer;

identifying the oldest packet that is written into said shared memory buffer;

identifying the availability of each of said plurality of output links;

starting at the identified oldest packet that is written into said shared memory buffer, examining said packets in said output buffer in the order that is dictated by said linked list to identify the oldest packet that is intended for an available output link; and

dispatching, to the respective available output link, the packet that is identified as the oldest packet that is intended for an available output link;

wherein linking said packets includes setting a NEXT pointer for each packet that identifies the next packet on said linked list.

45. (new) A system for managing packets in a shared memory buffer that serves a plurality of output links comprising:

a shared memory buffer, that serves a plurality of output links, for storing packets intended for said output links:

a buffer controller connected to manage packets stored within said shared memory buffer, said buffer controller including means for:

linking said packets into a linked list in the order that said packets are written into said shared memory buffer;

identifying the oldest packet that is written into said shared memory buffer;

identifying the availability of each of said plurality of output links;

starting at the identified oldest packet that is written into said shared memory buffer, examining said packets in said output buffer in the order that is dictated by said linked list to identify the oldest packet that is intended for an available output link; and

dispatching, to the respective available output link, the packet that is identified as the oldest packet that is intended for an available output link;

wherein linking said packets includes setting a NEXT pointer for each packet that identifies the next packet on said linked list.